

REMARKS

The foregoing Amendment After Final and the following Remarks are submitted in response to the Final Office Action issued on June 15, 2006 in connection with the above-identified patent application, and are being filed within the three-month shortened statutory period set for a response by the Final Office Action.

Claims 1-6, 8-14, and 16 remain pending in the present application. Claims 1 and 9 have been amended to more positively and clearly recite subject matter within the claims in response to a rejection under 35 USC § 112. Applicants respectfully submit that no new matter has been added to the application by the Amendment After Final. Applicants also respectfully submit that the present Amendment After Final should be entered inasmuch as the Amendment is not believed to add any new subject matter to the claims and therefore should not require any further searching, and also to place the claims in better form for an appeal, should such an appeal be necessary.

Applicants request reconsideration and withdrawal of the final rejection of the claims consistent with the following remarks.

Preliminarily, the Examiner has rejected 1 and 9 under 35 USC § 112, second paragraph, as being indefinite for the reason that the “the cluster ‘cluster’ having automatically switched processing . . . “ is deemed to be ambiguous. Accordingly, Applicants have amended claims 1 and 9 at least in part to more positively and clearly recite such subject matter. In particular, in amending such claims, such subject matter is now clearly recited as a series of positive method steps. Thus, Applicants respectfully request reconsideration and withdrawal of the § 112, second paragraph, rejection.

The Examiner has again rejected the claims under 35 USC § 103 as being obvious over Bruck et al. (U.S. Patent No. 6,801,949) in view of Brendel et al. (U.S. Patent No. 5,774,660) . Applicants respectfully traverse the § 103 rejection insofar as it may be applied to the claims as amended.

Independent claim 1 of the present application recites a method of connecting a client application at a computing device by way of a network access module (NAM) at the

computing device to a software server 'server' on one of a plurality of hardware servers on a cluster 'cluster', where the server is remote from the computing device. In the method, the NAM at the computing device receives 'cluster' and 'server' from the client application, sends a first request message to 'cluster' requesting first connection information for connecting to 'server', receives from 'cluster' a first reply message containing the requested first connection information, and connects the client application to 'server' as instantiated on a first server on 'cluster' based on the received first connection information. Once connected, the client application and 'server' at the first hardware server may transact business.

At some point thereafter, the first hardware server on 'cluster' fails, and 'cluster' in response automatically switches processing for 'server' from the failed first hardware server to an operating second hardware server on 'cluster'. Notably, 'cluster' does so without notifying the NAM at the computing device that 'server' is no longer at the failed first hardware server and has been switched to the second hardware server, and the connection of the client application and 'server' is thereby disrupted.

The NAM at the computing device itself then discovers that the connection of the client application and 'server' has been disrupted. As was previously pointed out, the NAM has no fail-over support from 'cluster', and thus cannot re-direct a request from the client from the failed first hardware server to the operating second hardware server. That is, 'cluster' has no front-end or the like that provides such fail-over support by re-directing such a request, so the NAM at the computing device must itself discover how to connect to 'server' as relocated to the second hardware server.

Thus, the NAM at the computing device sends a second request message from the computing device to 'cluster' requesting second connection information for connecting to 'server' at the second hardware server. Thereafter, the NAM at the computing device receives from 'cluster' a second reply message containing the requested second connection information, and connects the client application to 'server' as instantiated on the second hardware server on 'cluster' based on the received second connection information, wherein once again connected, the client application and 'server' at the second hardware server may again transact business.

Independent claim 9 recites subject matter similar to that recited in claim 1, albeit in the form of a computer-readable medium with computer-executable instructions thereon implementing the method.

Applicants once again point out that server availability in a clustered system is oftentimes increased by allowing the clustered system to automatically switch processing for an instance of a software server from a failed hardware server to a working hardware server. Thus, the working hardware server takes the place of the failed hardware server and restores database services to a client formerly accessing data from the failed hardware server. The Bruck reference in fact discloses an example of a cluster of hardware servers with a front-end that automatically switches processing from a failed hardware server to a working hardware server, specifically at column 27, line 23 through column 8, line 24. As shown in the Bruck reference, the front-end thereof switches by redirecting a request to the failed hardware server at an IP address thereof to a working hardware server at an IP address thereof. Essentially, the IP address of the failed hardware server acts as a flag to the front-end that the request must be re-directed.

However, if a cluster does not provide any such fail-over support, a request from a client to a failed server at the cluster will itself fail unless the client is itself provided with fail-over support. Clearly, the Bruck reference cannot teach or disclose such a client with fail-over support, especially inasmuch as the Bruck cluster does itself have a front-end that provides such fail-over support.

In the present application, a client application at a client computer can connect over a network to an instantiated server on a cluster by knowing the name of the cluster and the name of the hardware server upon which the instance resides. In the present invention, the client application provides such information to a network access module (NAM) that resides on the client computer, and the NAM employs such information to obtain mapping information from the cluster that provides a physical network end-point for the instance of the server 14 on the cluster.

Thus, the present invention is characterized (1) by the NAM at the client resolving such mapping information, as opposed to the front-end of the cluster, and (2) by the front-end of the cluster not having any fail-over support to re-direct a request from a failed server to a working server, among other things. Put simply, fail-over support must be

provided at the client if not provided at the cluster. Again, inasmuch as the Bruck reference teaches fail-over support at the cluster and not at the client, such Bruck reference cannot teach or disclose such a client with fail-over support.

As was previously pointed out, the Bruck reference discloses a load balancing server system with a front-end server layer between a network (such as the Internet) and multiple back-end servers. The front-end performs the fail-over support and also dynamic load balancing. In particular, when a hardware server fails, the Bruck front-end automatically shifts network traffic from the failed machine to an operational machine without interrupting operation of the server system. Thus, and as was pointed out above, the Bruck reference in teaching such front-end server and functionality actually teaches away from the present invention as recited in the claims.

More to the point, the Bruck cluster is disclosed as performing mapping services for a client at a front end, and thus does not in fact disclose a NAM at the client that performs such mapping services, as is required by the claims of the present application. Put simply, then, the Bruck reference does not disclose or even suggest any NAM at a computing device that ascertains connection information for connecting to a server at a cluster by performing the actions recited in claims 1 and 9. Instead, in the Bruck reference, such actions would be performed by the Bruck cluster itself, thus obviating the need for a NAM at the client in the manner recited in claims 1 and 9. Again, the present invention is for situations where the cluster does not itself perform such actions.

The Examiner once again points in the Bruck reference to the disclosure of a network interface card (NIC) at columns 7 and 8, and asserts that such NIC may be interpreted to be the recited NAM in that such NIC 'basically assists' in connecting a client to a server and performs the functions recited. However, Applicants respectfully point out here that the disclosed NIC to which the Examiner points is in fact a NIC at each of machines 302-308, which are part of the Bruck cluster. Thus, Bruck in fact discloses such NICs at the cluster and not at the client computer, as is required of the NAM recited in claims 1 and 9. For this reason alone, such a NIC cannot be interpreted to be the NAM of claims 1 and 9.

In addition, and as Applicants again point out, a NIC is nothing more than a conduit, and does not itself perform any substantive functionality. Thus, to state that a NIC 'basically assists' in connecting the client to the server is a severe generalization that is

unwarranted based even on the broadest interpretation of the Bruck reference. Moreover, Applicants respectfully point out that claims 1 and 9 recite the NAM performing several specific method steps, none of which a NIC performs, and most certainly none of which a NIC at the Bruck cluster performs. Quite simply, the claims of the present application require a NIC at a client computer performing fail-over functions to ensure that a client application remains connected to a software server at a cluster, and the Bruck reference teaches no such NAM at the client, and would not teach such a NAM at the client inasmuch as Bruck fail-over functions are performed by the Bruck cluster at the front-end thereof. Accordingly, Applicants again respectfully submit in this regard that the NIC of the Bruck reference cannot be interpreted to be the NAM of the computing device as recited in claims 1 and 9.

The Examiner concedes that although the Bruck reference discloses use of a cache, such cache is not at the computing device and is not used by a NAM at the computing device in the manner recited in the claims. Nevertheless, the Examiner argues that the Brendel reference discloses such a cache. That notwithstanding, Applicants merely point out that the Brendel reference like the Bruck reference fails to disclose or even suggest the NAM at the computing device functioning in the manner recited in claims 1 and 9.

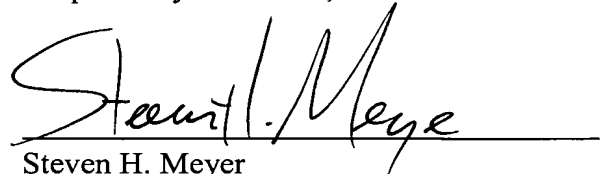
As a result, and for all of the aforementioned reasons, Applicants respectfully submit that the Bruck reference and the Brendel reference do not disclose or suggest the subject matter recited in independent claims 1 or 9 or any claims depending therefrom, including claims 2-6, 8, 10-14, and 16. Accordingly, and for all the aforementioned reasons, Applicants respectfully submit that the Bruck reference and the Brendel reference cannot be applied to make obvious such claims. Thus, Applicants respectfully request reconsideration and withdrawal of the § 103 rejection.

DOCKET NO.: MSFT-0688/180597.1
Application No.: 09/924,731
Office Action Dated: June 15, 2006

**PATENT
REPLY FILED UNDER EXPEDITED
PROCEDURE PURSUANT TO
37 CFR § 1.116**

In view of the foregoing discussion, Applicants respectfully submit that the present application, including claims 1-6, 8-14, and 16, is in condition for allowance, and such action is respectfully requested.

Respectfully submitted,

A handwritten signature in black ink, appearing to read "Steven H. Meyer", is written over a horizontal line.

Steven H. Meyer
Registration No. 37,189

Date: September 11, 2006

Woodcock Washburn LLP
One Liberty Place - 46th Floor
Philadelphia PA 19103
Telephone: (215) 568-3100
Facsimile: (215) 568-3439